

CLAIMS

1. A file server system, comprising:

two or more nodes, each node configured to run two or more virtual servers, each virtual server having as exclusive resources a virtual interface to clients and one or more file systems.

2. The system of claim 1, wherein the virtual interface comprises a virtual IP address.

3. The system of claim 1, wherein the virtual interface comprises two or more virtual IP addresses.

4. The system of claim 1, wherein clients access the file systems using NFS or CIFS protocols.

5. The system of claim 1, further comprising failover computer program instructions operable to be executed to cause the system to:

detect a failure of a first node; and

migrate each virtual server on the first node to a different node in the system.

6. The system of claim 5, wherein each virtual server has an associated failover priority, and the failover instructions further comprise instructions to:

migrate virtual servers in order of their respective priorities.

7. The system of claim 5, wherein the failover instructions further comprise instructions to:

recognize a virtual server that is identified as not to be migrated in the event of node failure and prevent migration of a so-identified virtual servers when it is on a node that fails.

8. The system of claim 1, further comprising rerouting computer program instructions operable to be executed to cause the system to:

detect a failure in a first subnet connected to a first node, the first node having a network connection to a first client;

5 identify a second node having a network connection to the first client and a connection over a second, different subnet to the first node;

use the second node as a router in response to the detected failure to route data between the first client and the first node.

9. The system of claim 8, wherein before failure in the first subnet, the connection
10 between the first client and the first node is through a first virtual IP address assigned to a port on the first node, the rerouting instructions further comprising instructions to:

migrate the first virtual IP address to a port on the second node connected to the second subnet.

10. The system of claim 1, further comprising failover computer program instructions
15 operable to be executed to cause the system to:

detect a failure of a physical port on a first node;

determine whether any other physical port on the first node is good;

migrate all virtual IP addresses associated with the failed physical port to a good physical port on the first node if there is such a good port; and

20 migrate all virtual IP addresses associated with the failed physical port along with all virtual servers attached to such virtual IP addresses to a different, second node if there is no such good port on the first node.

11. The system of claim 10, wherein the failed physical port is on a first subnet and the good physical port is on a different, second subnet.

12. The system of claim 1, wherein the system comprises load-balancing computer program instructions operable to be executed to cause the system to:

calculate a balanced distribution of the virtual server loads across the nodes of the system, excluding any failed nodes; and

perform load balancing by migrating one or more virtual servers from heavily loaded nodes to less heavily loaded nodes.

13. The system of claim 1, further comprising computer program instructions operable to be executed on a first node to:

determine a load on each physical port on the first node; and

redistribute the virtual interfaces on the first node among the physical ports of the first node for load balancing over the physical ports.

14. The system of claim 1, further comprising computer program instructions operable to be executed to cause the system to:

detect an inability on a first node to access of shared storage unit; and

in response to detection of the inability to access the shared storage unit, migrate all virtual servers containing file systems on the shared storage unit to an alternative node that can access the storage unit if such an alternative node exists in the system.

15. The system of claim 12, wherein the load-balancing instructions are further operable to determine a load on each virtual server.

16. The system of claim 12, wherein the load-balancing instructions are further operable to determine a load on each physical server.

17. The system of claim 12, wherein the nodes include a master node and the load-balancing instructions are operable to be executed on the master node.

18. The system of claim 12, wherein the load-balancing instructions are operable to migrate a first virtual server and a second virtual server from a first node, the first virtual server being migrated to a second node of the system and the second virtual server being migrated to a different, third node of the system.

19. The system of claim 12, wherein the load-balancing instructions are operable to balance system load as part of a failover process.

20. The system of claim 12, wherein the load-balancing instructions are operable to balance system load independent of any failover occurring.

21. The system of claim 1, further comprising computer program instructions operable to be executed to cause the system to:

detect without user intervention a file system degradation of a first file system;
and

block access to the first file system in response to the detection of the degradation,
repair the first file system, and then permit access to the first file system, all without user
intervention.

22. A file server system, comprising:

a node configured with a virtual server having two or more simultaneously active
virtual IP addresses.

23. The system of claim 22, wherein the node is configured with a second virtual server
having two or more other simultaneously active virtual IP addresses.

24. A file server system, comprising:

two or more nodes, each node being configured to run a virtual server having a
virtual IP address, and each node being configured with two or more physical ports;

wherein a first node is further configured to:

detect a failure of a physical port on the first node;

determine whether any other physical port on the first node is good;

migrate all virtual IP addresses associated with the failed physical port to a
good physical port on the first node if there is such a good port; and

migrate all virtual IP addresses associated with the failed physical port
along with all virtual servers attached to such virtual IP addresses to a different, second
node if there is no such good port on the first node.

25. A computer program product, tangibly stored on a computer-readable medium or propagated signal, for execution in multiple nodes of a file server system cluster, comprising instructions operable to cause a programmable processor to:

detect a failure of a first node of the cluster; and

5 migrate each of multiple virtual servers on the first node to a different node in the cluster.

26. The product of claim 25, further comprising instructions to:

migrate virtual servers in order of their respective priorities.

27. The product of claim 25, further comprising instructions to:

10 recognize a virtual server that is identified as not to be migrated in the event of node failure and prevent migration of a so-identified virtual servers when it is on a node that fails.

28. The product of claim 25, further comprising instructions to:

15 detect a failure in a first subnet connected to a first node, the first node having a network connection to a first client;

identify a second node having a network connection to the first client and a connection over a second, different subnet to the first node;

use the second node as a router in response to the detected failure to route data between the first client and the first node.

29. The product of claim 25, further comprising instructions to:

detect a failure of a physical port on a first node of the cluster;

determine whether any other physical port on the first node is good;

20 migrate all virtual IP addresses associated with the failed physical port to a good physical port on the first node if there is such a good port; and

25 migrate all virtual IP addresses associated with the failed physical port along with all virtual servers attached to such virtual IP addresses to a different, second node of the cluster if there is no such good port on the first node.

30. The product of claim 29, wherein before failure in the first subnet, the connection between the first client and the first node is through a first virtual IP address assigned to a port on the first node, the rerouting instructions further comprising instructions to:

migrate the first virtual IP address to a port on the second node connected to the second subnet.

31. The product of claim 25, further comprising load-balancing instructions to:

determine a load produced by each virtual server;

calculate a balanced distribution of the virtual server loads across the nodes of the server, excluding any failed nodes; and

perform load balancing by migrating one or more virtual servers from heavily loaded nodes to less heavily loaded nodes.

32. The system of claim 31, wherein the nodes include a master node and the load-balancing instructions are operable to be executed on the master node.

33. The system of claim 31, wherein the load-balancing instructions are operable to migrate a first virtual server and a second virtual server from a first node, the first virtual server being migrated to a second node of the system and the second virtual server being migrated to a different, third node of the system.

34. A computer program product, tangibly stored on a computer-readable medium or propagated signal, for execution in a node of a file server system cluster in which virtual servers have virtual IP addresses associated with physical ports, the product comprising instructions operable to cause a programmable processor to:

detect a failure of a physical port on a first node of the cluster;

determine whether any other physical port on the first node is good;

migrate all virtual IP addresses associated with the failed physical port to a good physical port on the first node if there is such a good port; and

migrate all virtual IP addresses associated with the failed physical port along with all virtual servers attached to such virtual IP addresses to a different, second node if there is no such good port on the first node.